2次元積符号用繰り返し型デコーダ設計仕様書

Design Wave 設計コンテスト 2006

デバイスの記事　ビギナーズ

和田知久

最近では携帯電話や地上デジタル放送など，移動体を対象とした無線通信技術の発達が著しく，またLSIの進歩に伴い，非常に複雑な信号処理を適用して無線通信システムの高性能化が実現されています．昨年度（2005年度）はディジタル方式のFMレシーバ回路を設計し，ディジタル・データの伝送を行いました．しかし，実際のディジタル通信ではエラー訂正技術を用いて，伝送されたディジタル・データに対してエラー訂正処理を行い，データ転送の信頼性を向上させています．最近，シャノンの理論限界に迫る高性能を実現するエラー訂正方式としてターボ符号や低密度パリティ検査符号が注目され，一部で実用化が始まっています．ターボ符号では送信側（エンコーダ）の構成は単純ですが，受信側（デコーダ）で同じデータに対して繰り返し処理を行い，エラー訂正能力を向上させています．

（筆者）

コンテストなので，小さめのディジタル回路を設計することを念頭に，なるべく簡単な回路構成要素を組み合わせて実現できなければなりません．ターボ符号は課題として規模が大きすぎるため，今回はターボ符号デコーダと似た繰り返し処理を用いる2次元積符号のデコーダ回路を設計します．送信情報はエンコーダで2×2の2次元配列に並べられます．その2行2列のそれぞれに対してパリティ・ビットを生成します．2行2列なので，全部で4ビットのパリティが生成されます．したがって，送信情報4ビットにパリティ4ビットが付加されて，8ビットが一つの処理単位となります．この8ビット情報がBPSK（binary phase shift keying）方式で変調され，送信され，ノイズの影響を受けます．受信側ではこのノイズの影響を受けた信号の復号，すなわちエラー訂正を行います．

HDL（VHDLもしくはVerilog HDL）による設計と論理合成を行います．FPGA向けに無償で提供されている設計ツールでも参加できます．また，余裕のある方はFPGAなどへ実装すれば，その努力を認められ，高い評価が得られると思います．FPGAへの実装にもぜひトライしてみてください．

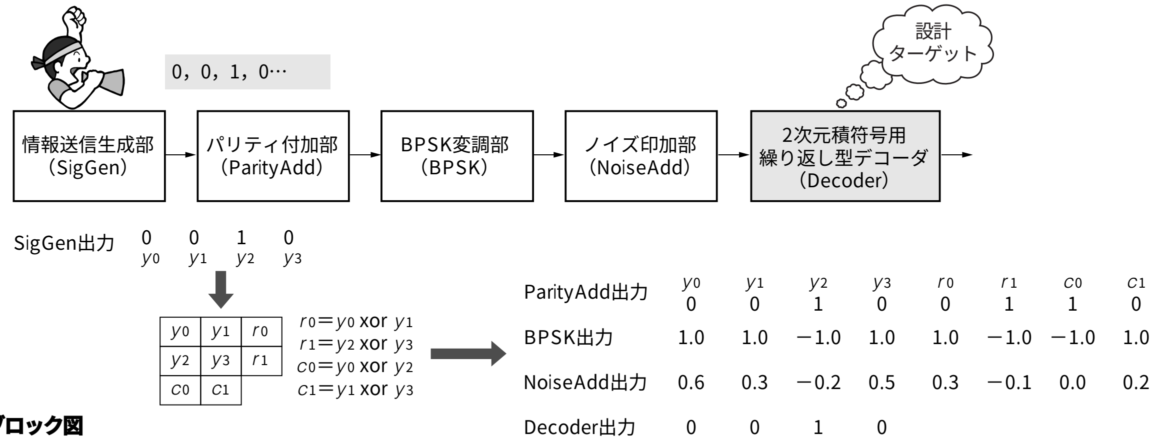


図1
システム・ブロック図

1．デコーダの構成

図1に今回想定するシステムのブロック図を示します．システムは大きく分けて，送信情報生成部（SigGen），そのSigGenの出力の情報ビットにパリティ検査ビットを付加するパリティ付加部（ParityAdd），'0'もしくは'1'のディジタル値をそれぞれ1.0，−1.0に変換するBPSK変調部（BPSK），実際のワイヤレス伝送時のノイズの印加に対応するノイズ印加部（NoiseAdd），そのノイズの印加されたBPSK信号を受信してエラー訂正を行う2次元積符号用繰り返し型デコーダ（Decoder）から構成されます．

図1を見るとわかりますが，4ビットの送信情報に対して，さらに4ビットのパリティ・ビットを付加して，8ビットに変換します．この8ビットはBPSKで8個の1.0もしくは−1.0の実数値に変換されます．通信路ではノイズの影響により，この8個の実数値が異なる値になります．デコーダでは8個の実数値を受け取り，これをもとにエラー訂正を行い，送信情報ビット4ビットの復号を行います．

● 2次元積符号（2 dimensional product code）

図2に2次元積符号の構成方法を示します．SigGen出力は4ビットごとにグループ分けされています．この4ビットを{y_0，y_1，y_2，y_3}とし，図のような2×2の正方行列を作成します．各行，各列に対して1ビットのパリティ検査ビットを付加します．行のパリティをそれぞれ，r_0，r_1，列のパリティをc_0，c_1とします．この四つのパリティ・ビットを元の4ビットの情報ビットに連結し，{y_0，y_1，y_2，y_3，r_0，r_1，c_0，c_1}なる8ビットの2次元積符号を生成します．

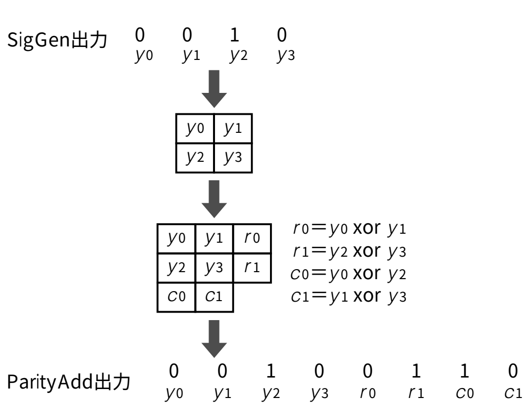


図2　2次元積符号

余談ですが，4ビット情報に対して送信ビットは8ビットなので，このような符号をコード・レート$R=1/2$（＝4ビット/8ビット）の符号と言います．

● BPSKとノイズ印加

実際に'0'，'1'からなるディジタル情報を無線で送信する場合には，電波の波を使って情報を伝送することになります．BPSK変調は，位相が0°と180°の異なる2種類の波を使って伝送する方式です．BPSKの説明は省略しますが，cos（0°）＝1.0，cos（180°）＝−1.0なので，ここではディジタル情報'0'に対して実数1.0を，ディジタル情報'1'に対して実数−1.0を伝送します．

このようすを**図3**の上半分に示します．1.0と−1.0の中間値は0.0なので，受信側では受信信号が0.0より上であれば送信データは'0'，0.0より下であれば'1'と復調することができます．

このBPSK変調された信号は電波として伝送され，受信機側に届く間にノイズが加えられます．ここではもっとも一般的に用いられているランダム・ノイズ（加法性白色ガウス・ノイズ）を印加します．ランダム・ノイズには特徴があり，ノイズを長時間にわたって平均すると0となります．また，ノイズの振幅はガウス分布に従い，小振幅のノイズの出現頻度が高く，大振幅のノイズの出現頻度は低くなります．

一般に，BPSKのような送信信号とノイズの割合を定義するために，信号ノイズ比（SN比またはS/N）が用いられ，ここでもSN比を用います．SN比を定義するためには，信号パワーとノイズ・パワーを求め，その比をとる必要があ

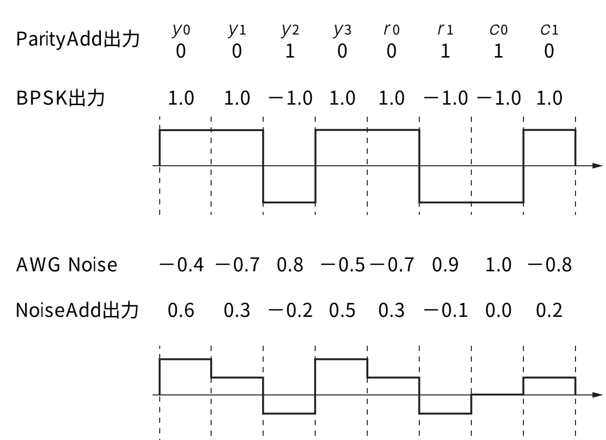


図3　BPSK変調とノイズ印加

ります．信号パワーは信号振幅の2乗の平均で求められます．今回のBPSKではつねに信号振幅は1.0もしくは－1.0なので，その2乗は1.0です．平均も1.0となり，信号パワーは1.0ということになります．

また，**図3**のAWG Noiseがノイズ振幅であり，振幅は{－0.4，－0.7，0.8，－0.5，－0.7，0.9，1.0，－0.8}なので，パワー平均は，

$$Noise\ Power = 1/8 \times (0.16 + 0.49 + 0.64 + 0.25 + 0.49 + 0.81 + 1.0 + 0.64) = 0.56$$

であり，

$$S/N = 1.0\ /\ 0.56 = 1.78$$

となります．一般にはこれを以下のようにdB(デシベル)値に変換します．

$$S/N(\mathrm{dB}) = 10 \cdot \log_{10}(1.78) = 2.5\ (\mathrm{dB})$$

したがって，ここではS/N＝2.5dBのノイズを印加したことになります．**図3**の最下段の波形はノイズ印加後の送信波形であり，この波形を受信側が復調します．受信信号{0.6，0.3，－0.2，0.5，0.3，－0.1，0.0，0.2}と中間値0.0を比較することで，対応するディジタル情報を復元できますが，受信信号＝0.0はちょうど中間値と同じなので，復元は困難です．しかし，この八つの受信信号は元は4ビットの情報で，冗長性があります．エラー訂正処理を行えば正しく復調することが可能であり，これが今回の設計テーマとなります．

2．復号アルゴリズム

● 事前確率と事後確率

今回は受信した8個の信号値R＝{R_{y0}，R_{y1}，R_{y2}，R_{y3}，R_{r0}，R_{r1}，R_{c0}，R_{c1}}から，元の4ビットのディジタル情報値{y_0，y_1，y_2，y_3}を推定します．例えば，y_0は‘0’か‘1’の二つの場合しかなく，8個の信号値Rを受信したとき，以下の二つの確率のどちらが大きいかを判断すればよいことになります．

1）あるRを受信したとき，送信情報y_0が‘0’である条件確率：$P(y_0=0|R)$

2）あるRを受信したとき，送信情報y_0が‘1’である条件確率：$P(y_0=1|R)$

Rを受信する以前は，$P(y_0=0)=P(y_0=1)=0.5$で同じ確率ですが，Rを受信したことにより，$P(y_0=0|R)$と$P(y_0=1|R)$の大小に差が生じたことになります．専門用語では，Rを受信する前の$P(y_0=0)$や$P(y_0=1)$を事前確率と呼び，Rを受信した後の$P(y_0=0|R)$や$P(y_0=1|R)$を事後確率と呼びます．

● 対数事後確率比(事後値)

上記二つの事後確率の大小比較を行う別の方法として，対数事後確率比(事後値)$\Lambda(y_0)$を定義します．

$$\Lambda(y_0) = \ln\left(\frac{P(y_0=0|R)}{P(y_0=1|R)}\right)$$

これは上記二つの事後確率の比の自然対数をとったものです．これにより，確率の大小関係の判断の代わりに，事後値が正か負かで判断できることになります．すなわち，$\Lambda(y_0)>0$であれば，$P(y_0=0|R)>P(y_0=1|R)$であり，Rを受信したときに，$y_0=0$と判断することができます．

ベイズの公式より，

$$P(y_0=0|R) = \frac{P(y_0=0,R)}{P(R)}$$

$$P(y_0=1|R) = \frac{P(y_0=1,R)}{P(R)}$$

と変形できるので，

$$\Lambda(y_0) = \ln\left(\frac{\dfrac{P(y_0=0,R)}{P(R)}}{\dfrac{P(y_0=1,R)}{P(R)}}\right) = \ln\left(\frac{P(y_0=0,R)}{P(y_0=1,R)}\right)$$

となります．

● 事後値＝通信路値＋事前値＋外部値の導出

ここで，$P(y_0=0,\ R)$の意味を考えると，送信情報が$y_0=0$であり，かつあるR＝{R_{y0}，R_{y1}，R_{y2}，R_{y3}，R_{r0}，R_{r1}，R_{c0}，R_{c1}}を受信した確率です．ここで行方向のパリティの関係に注目すると，$r_0=y_0$ xor y_1であるので，$y_0=r_0$ xor y_1の関係が成立し，送信情報y_0＝‘0’ならばr_0 xor y_1＝‘0’も成立していることになります．すなわち，以下のように変形可能です(xorは排他的論理和)．

$$\Lambda(y_0) = \ln\left(\frac{P(y_0 = 0, R)}{P(y_0 = 1, R)}\right)$$

$$= \ln\left(\frac{P(y_0 = 0, R_{y0}) \cdot P(r_0 \oplus y_1 = 0, \{R_{r0}, R_{y1}\})}{P(y_0 = 1, R_{y0}) \cdot P(r_0 \oplus y_1 = 1, \{R_{r0}, R_{y1}\})}\right)$$

自然対数内の積は自然対数項の加算に変形できるので，

$$\Lambda(y_0) = \ln\left(\frac{P(y_0 = 0, R_{y0})}{P(y_0 = 1, R_{y0})}\right) + \ln\left(\frac{P(r_0 \oplus y_1 = 0, \{R_{r0}, R_{y1}\})}{P(r_0 \oplus y_1 = 1, \{R_{r0}, R_{y1}\})}\right)$$

さらにこの1項目はベイズの公式により，

$$\ln\left(\frac{P(y_0 = 0, R_{y0})}{P(y_0 = 1, R_{y0})}\right) = \ln\left(\frac{P(R_{y0} | y_0 = 0) \cdot P(y_0 = 0)}{P(R_{y0} | y_0 = 1) \cdot P(y_0 = 1)}\right)$$

$$= \ln\left(\frac{P(R_{y0} | y_0 = 0)}{P(R_{y0} | y_0 = 1)}\right) + \ln\left(\frac{P(y_0 = 0)}{P(y_0 = 1)}\right)$$

と変形できるので，

$$\Lambda(y_0) = \ln\left(\frac{P(R_{y0} | y_0 = 0)}{P(R_{y0} | y_0 = 1)}\right) + \ln\left(\frac{P(y_0 = 0)}{P(y_0 = 1)}\right) + \ln\left(\frac{P(r_0 \oplus y_1 = 0, \{R_{r0}, R_{y1}\})}{P(r_0 \oplus y_1 = 1, \{R_{r0}, R_{y1}\})}\right)$$

となります．この意味を考えると，

1）第1項目は$y_0 = 0$を送って，あるR_{y0}を受信する確率と，$y_0 = 1$を送って，ある同じR_{y0}を受信する確率の対数比であり，これは送信チャネルのノイズの量，すなわちS/Nによって決定される値です．したがって，この1項目を通信路値L_{ch}と呼びます．

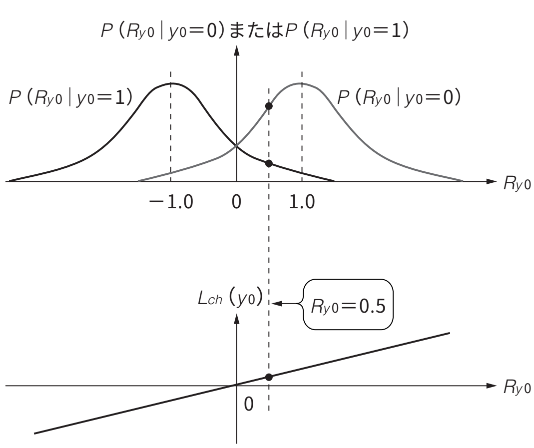


図4　白色ガウス・ノイズによる受信値の分散と通信路値

2）第2項目は送信情報のy_0が0である確率と1である確率の対数比であり，偏りのないデータを送信していると仮定すれば，第2項目は0です．これは，送信情報に関する項であり，事前値と呼びます．

3）第3項目はy_0ではない，行のパリティで関連するほかの部分の対数確率比に関する項であり，外部値L_eと呼びます．

すなわち，

事後値Λ ＝ 通信路値L_{ch} ＋ 事前値 ＋ 外部値L_e

で事後値が計算可能となります．

● 白色ガウス・ノイズ印加時の通信路値の計算

BPSK変調，すなわち論理値‘0’に対して信号1.0を，論理値‘1’に対して信号−1.0を送出し，そこに白色ガウス・ノイズが加わると，**図4**の上図に示すように1.0もしくは−1.0を中心に受信信号値が分散します．ガウス・ノイズなので，受信信号の分布は正規分布に従い，式で表すと以下のようになります．

1）送信信号＝1.0の場合

$$P(R_{y0} | y_0 = 0) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(R_{y0} - 1)^2}{2\sigma^2}\right)$$

2）送信信号＝−1.0の場合

$$P(R_{y0} | y_0 = 1) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(R_{y0} + 1)^2}{2\sigma^2}\right)$$

ここでσ^2はノイズの分散

これより，通信路値L_{ch}を求めると，

$$L_{ch}(R_{y0}) = \ln\left(\frac{P(R_{y0} | y_0 = 0)}{P(R_{y0} | y_0 = 1)}\right) = \frac{2R_{y0}}{\sigma^2}$$

となり，L_{ch}は受信信号R_{y0}に比例することになります．

● 外部値の計算

外部値は以下で表されます．

$$L_e = \ln\left(\frac{P(r_0 \oplus y_1 = 0, \{R_{r0}, R_{y1}\})}{P(r_0 \oplus y_1 = 1, \{R_{r0}, R_{y1}\})}\right)$$

$$= \ln\left(\frac{P(r_0 \oplus y_1 = 0 \mid \{R_{r0}, R_{y1}\})\, P(\{R_{r0}, R_{y1}\})}{P(r_0 \oplus y_1 = 1 \mid \{R_{r0}, R_{y1}\})\, P(\{R_{r0}, R_{y1}\})}\right)$$

$$= \ln\left(\frac{P(r_0 \oplus y_1 = 0 \mid \{R_{r0}, R_{y1}\})}{P(r_0 \oplus y_1 = 1 \mid \{R_{r0}, R_{y1}\})}\right)$$

r_0 xor $y_1 = 0$ となるのは，$r_0 = y_1 = 0$ もしくは$r_0 = y_1 = 1$となる場合であり，r_0 xor $y_1 = 1$となるのは$r_0 = 0$，$y_1 = 1$もしくは$r_0 = 1$，$y_1 = 0$となる場合なので，以下のように変形することができます．

$$L_e = \ln\left(\frac{P(r_0 = 0, y_1 = 0 \mid \{R_{r0}, R_{y1}\}) + P(r_0 = 1, y_1 = 1 \mid \{R_{r0}, R_{y1}\})}{P(r_0 = 1, y_1 = 0 \mid \{R_{r0}, R_{y1}\}) + P(r_0 = 0, y_1 = 1 \mid \{R_{r0}, R_{y1}\})}\right)$$

$$= \ln\left(\frac{P(r_0 = 0 \mid R_{r0})P(y_1 = 0 \mid R_{y1}) + P(r_0 = 1 \mid R_{r0})P(y_1 = 1 \mid R_{y1})}{P(r_0 = 1 \mid R_{r0})P(y_1 = 0 \mid R_{y1}) + P(r_0 = 0 \mid R_{r0})P(y_1 = 1 \mid R_{y1})}\right)$$

$$= \ln\left(\frac{1 + \dfrac{P(r_0 = 0 \mid R_{r0})}{P(r_0 = 1 \mid R_{r0})} \cdot \dfrac{P(y_1 = 0 \mid R_{y1})}{P(y_1 = 1 \mid R_{y1})}}{\dfrac{P(r_0 = 0 \mid R_{r0})}{P(r_0 = 1 \mid R_{r0})} + \dfrac{P(y_1 = 0 \mid R_{y1})}{P(y_1 = 1 \mid R_{y1})}}\right)$$

ここで，Λ_{r0}，Λ_{y1}を以下のように定義します．

$$\Lambda_{r0} = \ln\left(\frac{P(r_0 = 0 \mid R_{r0})}{P(r_0 = 1 \mid R_{r0})}\right) = \ln\left(\frac{P(R_{r0} \mid r_0 = 0)P(r_0 = 0)}{P(R_{r0} \mid r_0 = 1)P(r_0 = 0)}\right)$$

$$= \ln\left(\frac{P(R_{r0} \mid r_0 = 0)}{P(R_{r0} \mid r_0 = 1)}\right) + \ln\left(\frac{P(r_0 = 0)}{P(r_0 = 1)}\right)$$

$$= \ln\left(\frac{P(R_{r0} \mid r_0 = 0)}{P(R_{r0} \mid r_0 = 1)}\right) = \frac{2R_{r0}}{\sigma^2}$$

$$\Lambda_{y1} = \ln\left(\frac{P(y_1 = 0 \mid R_{y1})}{P(y_1 = 1 \mid R_{y1})}\right) = \frac{2R_{y1}}{\sigma^2}$$

すると，$|\Lambda_{r0}|$，$|\Lambda_{y1}|$の差が大きい場合には，以下のように近似することができます．

$$L_e = \ln\left(\frac{1 + \exp(\Lambda_{r0} + \Lambda_{y1})}{\exp(\Lambda_{r0}) + \exp(\Lambda_{y1})}\right)$$

$$\approx \mathrm{sgn}(\Lambda_{r0} \cdot \Lambda_{y1}) \min(|\Lambda_{r0}|, |\Lambda_{y1}|)$$

$\mathrm{sgn}(\Lambda_{r0} \cdot \Lambda_{y1})$は$\Lambda_{r0}$と$\Lambda_{y1}$の積の符号であり，正であれば1.0，負であれば－1.0を返す関数とします．$\min(|\Lambda_{r0}|, |\Lambda_{y1}|)$は$\Lambda_{r0}$，$\Lambda_{y1}$の絶対値の小さいほうを返す関数です．

回路を構成するときは，Λ_{r0}，Λ_{y1}を変数として用いて，上記近似式を計算することになります．

3．繰り返し型の復号処理

復号処理に関するアルゴリズムの基本式を導出しましたが，ここでは結果的にどのように復号するのかを説明します．

手順1：$R = \{R_{y0}, R_{y1}, R_{y2}, R_{y3}, R_{r0}, R_{r1}, R_{c0}, R_{c1}\}$からなる八つの実数値を受信

この8個の値から通信路値，外部値などを計算しますが，通信路値のL_{ch}は受信値に対して$2/\sigma^2$を乗算し，外部値のΛの計算も同じく$2/\sigma^2$を乗算する必要があります．結果的にすべての計算値に$2/\sigma^2$が乗算され，事後値の正か負かの判断には影響しないため，ここでは$2/\sigma^2 = 1$と簡単化して話を進めます．

手順2：通信路値から行のパリティの性質を使って外部値を計算

図5の最上段に示すように，通信路値L_{ch}と事前値prioriの加算を行います．ただし，初期であるので事前値はすべて0.0です．この和に対して，外部値L_eを計算します．外部値L_eは行パリティの関係と列パリティの関係から計算できますが，まず行パリティの関係を用いて以下のように計算します．

$$y_{0e}(0) = \mathrm{sgn}(R_{y1} \cdot R_{r0}) \cdot \min(|R_{y1}|, |R_{r0}|)$$
$$y_{1e}(0) = \mathrm{sgn}(R_{y0} \cdot R_{r0}) \cdot \min(|R_{y0}|, |R_{r0}|)$$
$$y_{2e}(0) = \mathrm{sgn}(R_{y3} \cdot R_{r1}) \cdot \min(|R_{y3}|, |R_{r1}|)$$
$$y_{3e}(0) = \mathrm{sgn}(R_{y2} \cdot R_{r1}) \cdot \min(|R_{y2}|, |R_{r1}|)$$

実際にはこの外部値L_eを次の繰り返しにおいて事前値に使用するだけなので，事後値posterioriの計算は不要ですが，図のように事後値を毎回計算してもかまいません．

手順3：上記外部値を次の繰り返しの事前値として使用し，列パリティの性質を使って外部値を計算する

図5の上から2段目に示すように，以下のように次の外部値L_eを計算します．

$$y_{0e}(1) = \mathrm{sgn}\{(R_{y2} + y_{2e}(0)) \cdot R_{c0}\}$$

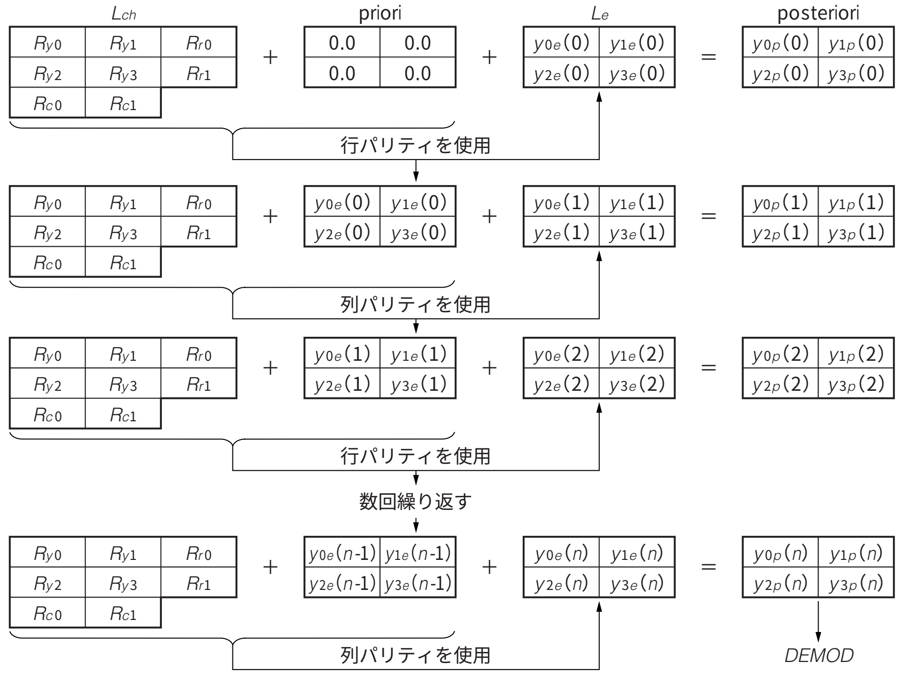


図5
繰り返し型の復号処理

$$\cdot \min\{|R_{y2}+y_{2e}(0)|,\ |R_{c0}|\}$$

$$y_{1e}(1)=\mathrm{sgn}\{R_{y3}+y_{3e}(0))\cdot R_{c1}\}$$
$$\cdot \min\{|R_{y3}+y_{3e}(0)|,\ |R_{c1}|\}$$

$$y_{2e}(1)=\mathrm{sgn}\{(R_{y0}+y_{0e}(0))\cdot R_{c0}\}$$
$$\cdot \min\{|R_{y0}+y_{0e}(0)|,\ |R_{c0}|\}$$

$$y_{3e}(1)=\mathrm{sgn}\{(R_{y1}+y_{1e}(0))\cdot R_{c1}\}$$
$$\cdot \min\{|R_{y1}+y_{1e}(0)|,\ |R_{c1}|\}$$

手順4：上記のように行パリティ，列パリティを用いた外部値計算を数回繰り返す

行パリティを用いた外部値の計算の一般式は，

$$y_{0e}(n-1)=\mathrm{sgn}\{(R_{y1}+y_{1e}(n-2))\cdot R_{r0}\}$$
$$\cdot \min\{|R_{y1}+y_{1e}(n-2)|,\ |R_{r0}|\}$$

$$y_{1e}(n-1)=\mathrm{sgn}\{(R_{y0}+y_{0e}(n-2))\cdot R_{r0}\}$$
$$\cdot \min\{|R_{y0}+y_{0e}(n-2)|,\ |R_{r0}|\}$$

$$y_{2e}(n-1)=\mathrm{sgn}\{(R_{y3}+y_{3e}(n-2))\cdot R_{r1}\}$$
$$\cdot \min\{|R_{y3}+y_{3e}(n-2)|,\ |R_{r1}|\}$$

$$y_{3e}(n-1)=\mathrm{sgn}\{(R_{y2}+y_{2e}(n-2))\cdot R_{r1}\}$$
$$\cdot \min\{|R_{y2}+y_{2e}(n-2)|,\ |R_{r1}|\}$$

列パリティを用いた外部値計算の一般式は，

$$y_{0e}(n)=\mathrm{sgn}\{(R_{y2}+y_{2e}(n-1))\cdot R_{c0}\}$$
$$\cdot \min\{|R_{y2}+y_{2e}(n-1)|,\ |R_{c0}|\}$$

$$y_{1e}(n)=\mathrm{sgn}\{(R_{y3}+y_{3e}(n-1))\cdot R_{c1}\}$$
$$\cdot \min\{|R_{y3}+y_{3e}(n-1)|,\ |R_{c1}|\}$$

$$y_{2e}(n)=\mathrm{sgn}\{(R_{y0}+y_{0e}(n-1))\cdot R_{c0}\}$$
$$\cdot \min\{|R_{y0}+y_{0e}(n-1)|,\ |R_{c0}|\}$$

$$y_{3e}(n)=\mathrm{sgn}\{(R_{y1}+y_{1e}(n-1))\cdot R_{c1}\}$$
$$\cdot \min\{|R_{y1}+y_{1e}(n-1)|,\ |R_{c1}|\}$$

筆者の経験では，5～6回繰り返せば十分のように思います．

手順5：最後に事後値を計算し，復号を行う

事後値posterioriが正であるなら‘0’であり，負であれば‘1’に復号することができます．

4．回路構成の例

これまで，復号アルゴリズムや繰り返しの復号処理を説明してきました．けっこう複雑に思われたかと思います．しかし，実際に回路にしてみると意外に単純な回路構成で実現することができます．

図6は回路構成例で，**図7**はその動作タイミング図です．図では見やすくするためにクロック（CLK）の配線が省略されていますが，各フリップフロップに接続されます．

図7に示すように，4ビットの送信データsenddataに対

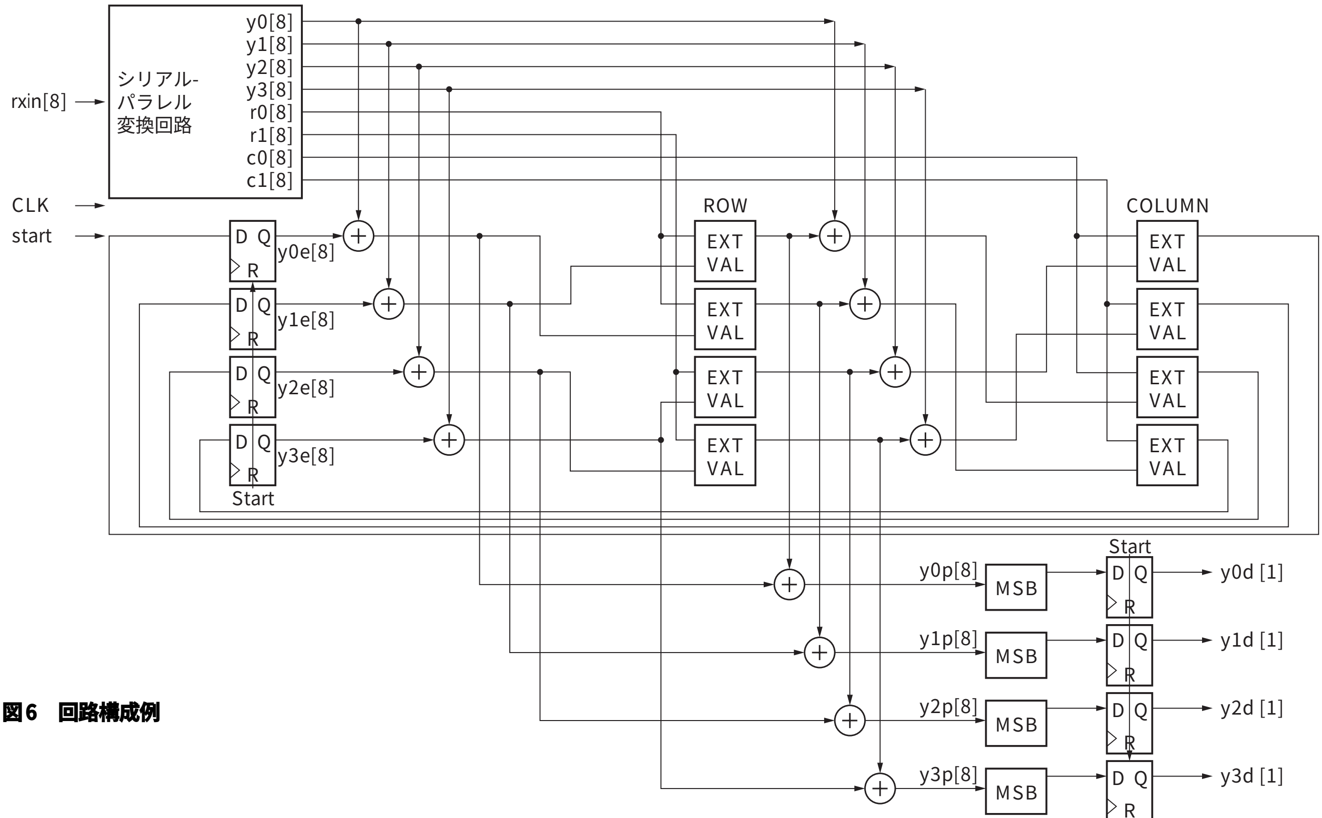


図6　回路構成例

して4ビットのパリティが後につなげられ，BPSK変調後送信され，ノイズの加えられた信号が8ビットのrxinに入力されます．送信単位（シンボル）の先頭を示すために，start信号も入力されます．8サイクルで，{R_{y0}, R_{y1}, R_{y2}, R_{y3}, R_{r0}, R_{r1}, R_{c0}, R_{c1}}の八つの実数値が受信されます．ここでは2の補数表現の8ビットを用いています．

シリアル-パラレル変換回路で，八つの8ビット値がパラレル信号{y0[8]，y1[8]，y2[8]，y3[8]，r0[8]，r1[8]，c0[8]，c1[8]}に変換されます．

start信号をリセット信号として用いて，外部値{y0e[8]，y1e[8]，y2e[8]，y3e[8]}の初期値がリセットされます．各サイクルごとに行パリティを使った外部値を計算し，その外部値を事前値として列パリティを使った外部値が計算され，外部値が更新されます．

同時に事後値{y0p[8]，y1p[8]，y2p[8]，y3p[8]}も更新され，8サイクル目の事後値のMSB（符号ビット）をstart信号で取り出すことにより，復号信号{y0d[1]，y1d[1]，y2d[1]，y3d[1]}を出力します．

図6中のEXTVALブロックはこれまでも説明したように，入力A，Bに対して，

$$out = sgn(A \cdot B) \cdot \min(|A|, |B|)$$

を計算するけっこう単純な組み合わせ回路ブロックです．

5．課　題

● LEVEL1 ：基本課題

基本課題では，**図7**に示すように，シリアルにやってくるrxinとstart信号をもとに復号出力{y0d[1]，y1d[1]，y2d[1]，y3d[1]}を出力します．**表1**に入出力信号とビット幅を示します．**図7**と**表1**は一つの例であり，同じにする必要はありません．自由に変更してください．

基本課題では，シリアルにやってくるrxinとstart信号から復号回路を設計し，S/N＝0dB，3dB，6dB，9dBの各条件における10000ビットの送信データ中のエラー訂正に失敗したデータ数をレポートしてください．

20000サイクル分のstart信号，および8ビットのrxin信号を本コンテストのホームページ（http://www.cqpub.co.jp/dwm/contest/）からダウンロードできます．コード・レートが1/2なので，実際には10000ビットに対応する送信信号です．rxinに対しては　S/N＝0dB，3dB，6dB，9dB，100dBを用意しており，S/N＝100dBはほとんどノ

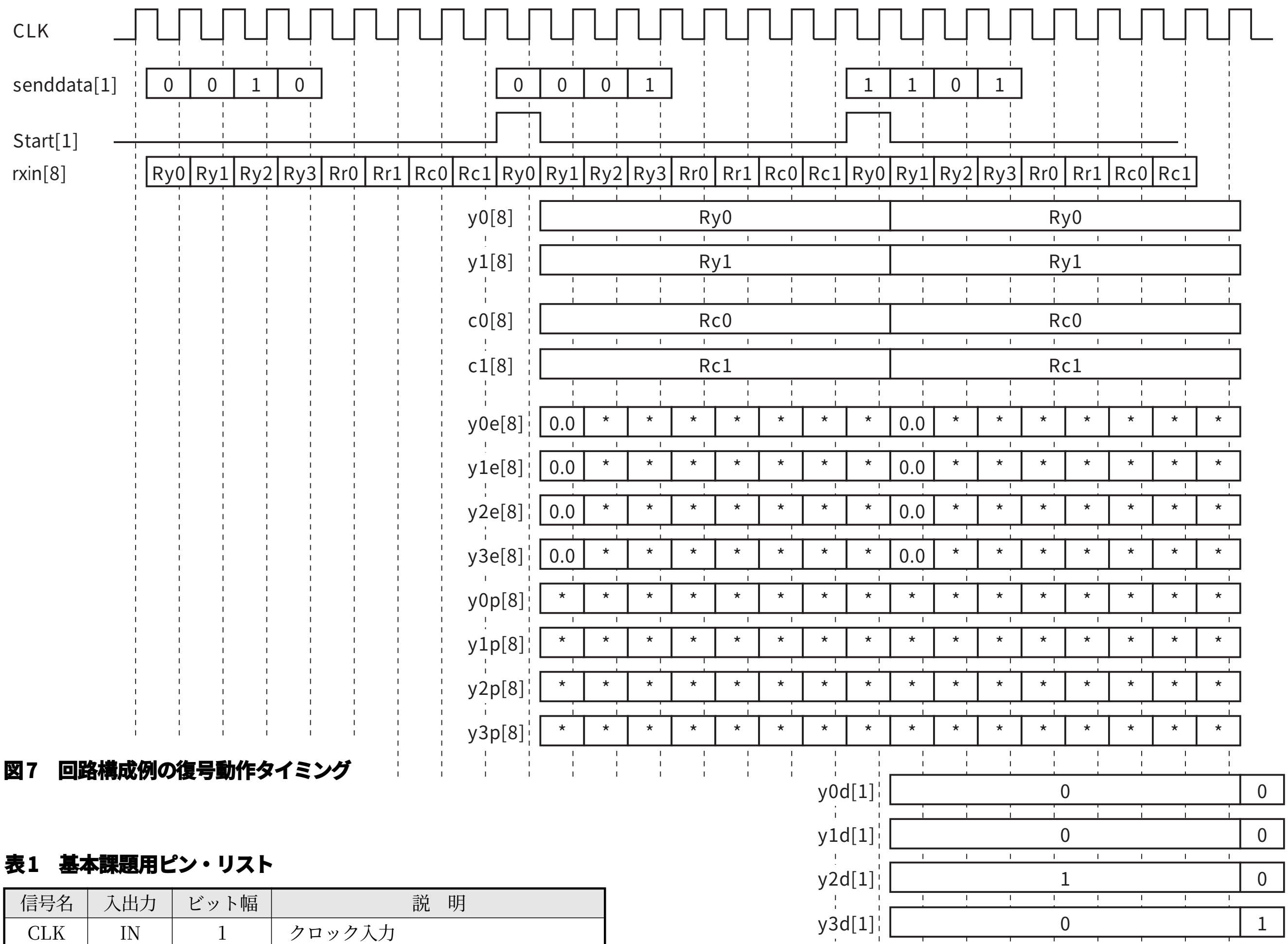


図7　回路構成例の復号動作タイミング

表1　基本課題用ピン・リスト

信号名	入出力	ビット幅	説　明
CLK	IN	1	クロック入力
start	IN	1	‘1’でシンボルの先頭を示す
rxin	IN	8	2の補数表現の送信信号
y0d	OUT	1	復調出力
y1d	OUT	1	
y2d	OUT	1	
y3d	OUT	1	

イズのない送信波形です．

rxinは2の補数表現の8ビット・データで，－128から127までの数値を表現できます．ここでは，BPSKのノイズなしの送信値を＋32，－32としています．

それぞれに対して復号動作を行い，10000ビットの送信信号に対して復号できなかったエラー・ビット数を報告してください．正確な10000ビットの送信データsenddataも本コンテストのホームページからダウンロードできます．

初期の40サイクル程度の波形図を**図8**に示します．最上段はstart信号であり，それ以下はS/N＝0dB，3dB，6dB，9dB，100dBの各条件におけるrxin波形を示しています．

最下段はノイズのない信号なので，BPSK信号を示しています．また，この表示では，8ビットの数を右へ7ビット・シフトした値で表示しているので，BPSKの振幅は0.25，－0.25と表示されています．

● LEVEL2：上級課題

上級編では送信データとして，上述の20000サイクル（送信データ10000ビット）の復号ができれば，どのような構成でもOKです．ユニークな方式をいろいろと考えてみてください．

案としては，

1）データは8ビットで与えられているが，回路としては8ビットすべてを使用する必要はなく，任意の精度で設計することにより高速化，省面積化を実現できるかもしれない

2）データの入力がシリアルでなく，自由な構成にすることで，おもしろいアーキテクチャを実現できるかもし

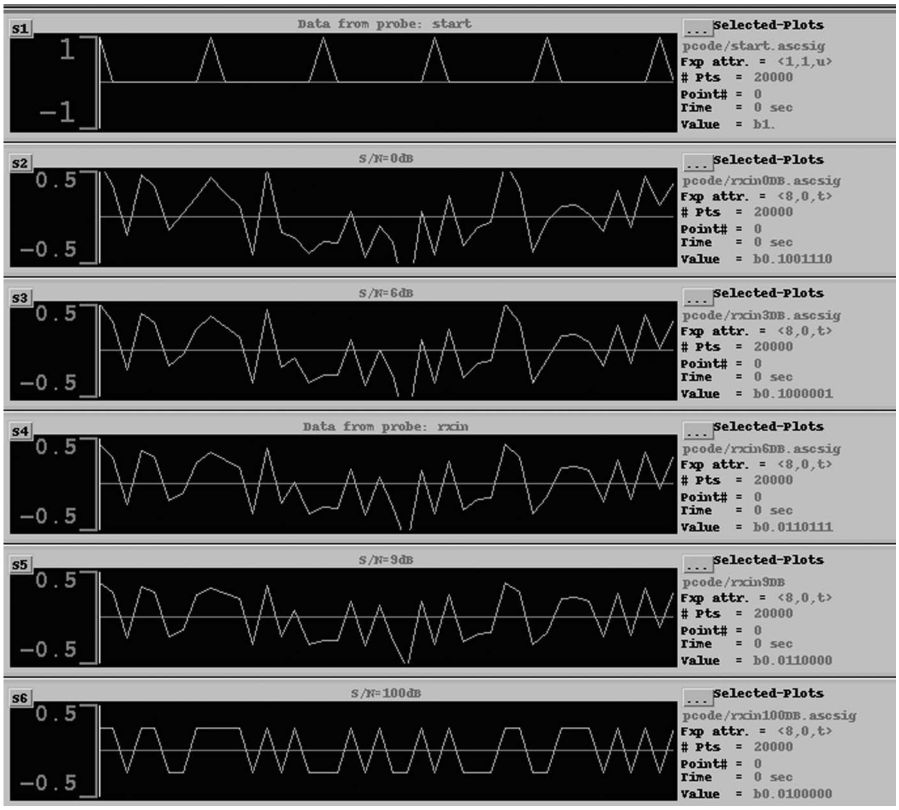


図8
入力波形図

れない

3）EXTVALの計算において上記説明では近似を行っているが，回路をつぎ込んで計算することにより，エラー訂正能力を向上できるかもしれない

4）今回，繰り返し型のデコード方式について説明したが，この方法をまったく無視して，rxinの受信値から独自の方法でデコードできるかもしれない

などが考えられます．

● 社会人部門の条件

社会人部門は，課題をより実践的にするため，FPGAに実装することを前提として設計していただきます．

レポートには，FPGA設計ツールが出力したレポート・ファイル（使用論理ブロック数や最大動作周波数がわかるもの）を添付してください．ターゲットFPGAの選択についても審査対象とします．設計意図，アーキテクチャ，コスト・パフォーマンスなどを考慮して，ターゲットFPGAを選択してください．

VHDL，Verilog HDL以外の設計言語による参加も認めます．

なお，2005年12月10日までの間に，弊社製FPGA評価キットの貸し出しや応募時のレポート方法に関する情報の提供などを行う場合があります．本コンテストのホームページや本誌2005年12月号～2006年2月号に掲載する「コンテスト参加要領」をかならずご確認ください．

6. 速度と規模の単位

筆者の勤務する大学では，論理合成用のツールとして米国Synopsys社の「Design Compiler」を使用していますが，このツールはだれもが使えるわけではありません．そこで，

- 2入力XORゲート一つの遅延時間を1UNIT_DELAY
- 2入力XORゲート一つの面積を1UNIT_AREA

とします．

具体的には，50入力XOR回路を合成していただき，その1段当たりの遅延時間を単位時間として速度の単位とします．50入力XOR回路のVHDLソース・コードを**リスト1**に示します（このファイルも本コンテストのホームページからダウンロード可能）．

Design Compilerでは，6段，49個のXOR回路が合成されます．クリティカル・パス遅延はreport_timingコマンドにより7.17であることがわかります．そこで7.17/6＝

リスト1　50入力XOR回路のVHDLソース・コード

```
library  IEEE;
use IEEE.STD_LOGIC_1164.all, IEEE.NUMERIC_STD.all;

entity PARITY is
    port ( A : in  unsigned(49 downto 0);
           Y : out std_logic );
end PARITY;

architecture RTL of PARITY is
begin

    process(A)
    variable TMP : std_logic;
    begin
        TMP := '0';
        for i in 0 to 49 loop
            TMP := TMP xor A(i);
        end loop;

      Y <= TMP;
    end process;
end RTL;
```

1.195を単位(1UNIT_DELAY)とします．例えば，ある遅延が20ならば，20/1.195＝17.74UNIT_DELAYということになります．面積はreport_areaコマンドのtotal cell areaにより147.0であることがわかります．XORゲート数は49個なので，147.0/49＝3.0を単位(1UNIT_AREA)とします．例えば，ある回路面積が200ならば，200/3.0＝66.67 UNIT_AREAとします．

7．提出レポートについて

レポートには以下の内容を含めてください．また，ページ数は少なめに，コンパクトにまとめてください．

<表紙>

1）代表者の氏名，チーム名，社会人/大学院修士/大学学部生/高専生の区別
2）共同設計者全員の名まえ(最大3名まで)，会社/学校名(学籍番号，学年)，住所，電話番号，E-mailなどの連絡先
3）取り組んだ課題(レベル1/レベル2)

<内容>

1）設計した回路ブロックの構成説明(ブロック図と説明)
2）設計した回路ブロックの動作説明
(動作波形図やパイプライン動作などの説明)
3）くふうした点，オリジナリティを出した点
(アピールが重要！)
4）クリティカル・パスの速度，論理合成後の回路規模，実現した*SN*比とエラー数の関係
5）VHDLもしくはVerilog HDLのコード
6）正常動作しているVHDL/Verilog HDLシミュレーションの波形
7）そのほか自由意見など

レポートはPDFファイルを推奨します．PDFファイルを作成できない場合はご相談ください．

社会人か学生かによって評価の方法が異なります．それにともない，レポートの送付先が異なります．

社会人部門：contest.dwm@cqpub.co.jp
学生部門　：wada@ie.u-ryukyu.ac.jp

締め切りは2006年1月27日(金)必着です．

8．審査のポイント

速度，回路規模だけでなく，アーキテクチャのユニークさ，アイデア，おもしろさを十分考慮して審査します(ちゃんとアピールしてね！)．

社会人は，Design Wave設計コンテスト審査委員会が審査を行います．学生は，琉球大学で1次審査を行い，最終審査に残ったチームは，2006年3月17日に沖縄産業支援センターで開催予定の発表会に招待され，その会場で最終審査を行います．大学院修士，学部生，高専生のレベルに応じて審査します．

仕様書に従ってまじめに作るのもけっこうですが，おもしろいアイデアを歓迎します．他人と違ったことをしよう！ 仕様の部分変更など，柔軟に受け付けます．

ENJOY HDL!　沖縄で会おう！

＊　＊　＊

本設計コンテストの学生部門は，主催：LSIコンテスト実行委員会，共催：琉球大学工学部 情報工学科，沖縄産業振興センター，九州半導体イノベーション協議会，協賛：ソニーLSIデザインにより実施されています．

参考・引用*文献

(1) 萩原春生；ターボ符号の基礎，第3章，トリケップス，1999年．

わだ・ともひさ
琉球大学工学部 情報工学科 教授